1. 数据结构及其分析；

通过分析文件字段关系，我们完成了岗位推荐数据的字段描述和关系说明的整理，通过对表格间主外键关系的分析，我们更好地了解整个数据结构。

1. 数据处理（算法实施）；

第一阶段：根据算法组需求将不同文件中求职者的信息汇总。因为不同表格中有求职者不同的信息（部分可用），企图将同一个人的所有信息汇总到一张表格上，方便对不同信息的取用。做法：利用matlab对表格中fk\_id（全局编号）做人的对应和汇总。

我们与高珩同学所负责的算法组进行交流后，根据提出的阿里匹配算法的数据要求，对新致提供的数据进行了结构调整以适应算法所需代码，得到结果table1.xlsx、table2.xlsx。按table1要求对ttyc\_personnel.xlsx进行处理，结果：personnel-table1.xlsx。过程中遇到了数据缺漏较多、ID匹配有误等问题。并对长文本进行了文本切割，也遇到了信息缺漏的问题。由于该算法实现存在困难，因此该数据最终未被采用。

第二阶段：（处理4.30-6.7的新数据）

字段描述与关系说明：我们对4.30-6.7的数据进行分析，并做出了字段描述，找到了各表之间的关系，并指明了主键ID与外键的对应关系。

正负样本处理：我们根据候选人表中的candidate state状态，来指定判断候选人是否通过初筛方法。

根据算法组要求汇总长文本信息。包括应聘者的个人优势描述，个人项目经历中的经历介绍，责任描述，以及有关岗位的职业描述和职业要求。其中应聘者普遍项目经历丰富，同一个人有多段项目经历，对信息进行字符串合并作为其总的项目经历的长文本信息

1. 问题及其分析

第一阶段：非结构性文本要求处理 'ttyc\_personnel\_work\_experience'中的EXPERIENCE\_DESCRIBE = Column(Text, comment='描述')一栏但是ttyc\_personnel与ttyc\_personnel\_work\_experience中的求职者无法对应，后来发现是标签解释不明，personnel中的id对应其他表格中的FK\_ID

第二阶段：1/候选人表和人才表重合度100%

2/某些行数据错位：是由于从数据库导出的文本字段中有引号，直接读取csv数据会错位，因为错位数据较少，我们本来计划删除这些数据，发现原因后，我们处理了csv字段中的引号的转义问题，从而解决了错位。